Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	58510	cluster$3 and (comput$3 server$2 client$2)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:13
S2	11409919	@ad<"20010313"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:14
S3	25927	S1 and S2	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:14
S4	5603991	improv$3 enhanc$3 upgrad$3 increas$3 rais$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:14
S5	6674503	performance efficiency operation capacity speed power processing	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:15
S6	1476369	S4 near3 S5	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:16
S7	11825	S3 and S6	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:16
S8	22546	bottleneck	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:16
S9	848	S7 and S8	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:16
S10	142533	data with analysis	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:17
S11	236	S10 and S9	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:17
S12	6416384	obtain$3 sampl$3 receiv$3 monitor$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:17

S13	985977	data with S12	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:17
S14	208	S11 and S13	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:19
S15	48912	S10 same S13	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:19
S16	65	S14 and S15	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:19
S17	57604	"international business machines". as.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 11:19
S18	64	S16 not S17	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:00
S19	2759	709/201.ccls. 709/205.ccls.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:00
S20	11409919	@ad<"20010313"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:00
S21	1619	S20 and S19	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:01
S22	57604	"international business machines". as.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:01
S23	1350	S21 not S22	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:01
S24	22546	bottleneck	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:01

S25	123	S24 and S23	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:02
S26	8567	node with sampl$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:02
S27	3	S25 and S26	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:31
S28	50	("6338117" "5659687" "5862332" "6894991" "5940870" "6119143" "5893086" "5907849" "5940841" "5940838" "5946686" "5950199" "5956734" "5960446" "5963963" "5974424" "5987477" "5999976" "6021508" "6023706" "6029168" "6915212" "6173420" "4947386" "4970722" "5394555" "6247077" "5379440" "5845071" "6049889" "6055599" "6122629" "6151688" "6247139" "6249879" "4939724" "5253161" "5404451" "5425136" "5497334" "5572674" "5588152" "5590345" "5625836" "5708836" "5710935" "5717943" "5734921" "5765011" "5794059").pn.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:31
S29	49	("5842031" "5963746" "5966528" "6058400" "6069986" "6094715" "6192483" "6199179" "6226734" "6226734" "6247109" "6260068" "6295585" "6332180" "6338112" "6353898" "6381682" "6434626" "6658478" "6922791" "6934768" "6292830" "5926632" "5884313" "5940831" "6049808" "6446061" "6560632" "5974574" "6735220" "6078990" "6081812" "6148349" "5905869" "4432057" "4903305" "5634125" "5652751" "5668943" "5687369" "5784697" "5796999" "5819083" "6028914" "6189111" "6453426" "6742072" "6807158" "6536012" "5727218").pn.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:31
S30	99	S28 S29	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 16:31

S31	59	S30 not S22	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S32	1	"20040162901"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/04 18:01
S33	326326	predict$4 forecast$4	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S34	58552	network with management	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S35	22992	bottleneck$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S36	212	S33 with S35	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S37	55	S34 and S36	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S38	11410536	@ad<"20010313"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:47
S39	26	S38 and S37	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 16:48
S40	326326	predict$4 forecast$4	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57
S41	58552	network with management	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57
S42	22992	bottleneck$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57

S43	212	S40 with S42	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57
S44	55	S41 and S43	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57
S45	11410536	@ad<"20010313"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57
S46	26	S45 and S44	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/11/15 17:57

Terms used **network management** and **bottleneck** and **cluster** Found **44,527** of **166,357**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 FlowMate: scalable on-line flow clustering

Ossama Younis, Sonia Fahmy

April 2005 **IEEE/ACM Transactions on Networking (TON)**, Volume 13 Issue 2

Publisher: ACM Press

Full text available: pdf(752.94 KB) Additional Information: full citation, abstract, references, index terms

We design and implement an efficient on-line approach, FlowMate, for clustering flows (connections) emanating from a busy server, according to shared bottlenecks. Clusters can be periodically input to load balancing, congestion coordination, aggregation, admission control, or pricing modules. FlowMate uses in-band (passive) end-to-end delay measurements to infer shared bottlenecks. Delay information is piggybacked on feedback from the receivers, or, if impossible, TCP or application round-trip t ...

Keywords: TCP, coordinated congestion management, load balancing, network monitoring, network tomography, shared bottleneck inference

2 Semantic search: Disambiguating Web appearances of people in a social network

Ron Bekkerman, Andrew McCallum

May 2005 **Proceedings of the 14th international conference on World Wide Web**

Publisher: ACM Press

Full text available: pdf(216.16 KB) Additional Information: full citation, abstract, references, index terms

Say you are looking for information about a particular person. A search engine returns many pages for that person's name but which pages are about the person you care about, and which are about other people who happen to have the same name? Furthermore, if we are looking for multiple people who are related in some way, how can we best leverage this social network? This paper presents two unsupervised frameworks for solving this problem: one based on link structure of the Web pages, another using ...

Keywords: Web appearance, document clustering, information bottleneck, link structure, name disambiguation, social network

3 STORM: lightning-fast resource management

Eitan Frachtenberg, Fabrizio Petrini, Juan Fernandez, Scott Pakin, Salvador Coll

November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**

Publisher: IEEE Computer Society Press

Full text available: pdf(324.44 KB) Additional Information: full citation, abstract, references, citings, index terms

Although workstation clusters are a common platform for high-performance computing

(HPC), they remain more difficult to manage than sequential systems or even symmetric multiprocessors. Furthermore, as cluster sizes increase, the quality of the resource-management subsystem---essentially, all of the code that runs on a cluster other than the applications---increasingly impacts application efficiency. In this paper, we present STORM, a resource-management framework designed for scalability and pe ...

4 A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks

Beongku An, Symeon Papavassiliou

November 2001 **International Journal of Network Management**, Volume 11 Issue 6

Publisher: John Wiley & Sons, Inc.

Full text available: pdf(162.46 KB) Additional Information: full citation, abstract, references, citings, index terms

In our proposed structure, mobile nodes are organized into nonoverlapping clusters which have adaptive variable-sizes according to their respective mobility. The mobility-based clustering (MBC) approach we are proposing uses a combination of both physical and logical partitions of the network (i.e. geographic proximity and functional relation between nodes, such as mobility pattern etc.)

5 Process migration

September 2000 **ACM Computing Surveys (CSUR)**, Volume 32 Issue 3

Publisher: ACM Press

Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Process migration is the act of transferring a process between two machines. It enables dynamic load distribution, fault resilience, eased system administration, and data access locality. Despite these goals and ongoing research efforts, migration has not achieved widespread use. With the increasing deployment of distributed systems in general, and distributed operating systems in particular, process migration is again receiving more attention in both research and product development. As hi ...

Keywords: distributed operating systems, distributed systems, load distribution, process migration

6 Affinity-based management of main memory database clusters

Minwen Ji

November 2002 **ACM Transactions on Internet Technology (TOIT)**, Volume 2 Issue 4

Publisher: ACM Press

Full text available: pdf(553.96 KB) Additional Information: full citation, abstract, references, index terms

We study management strategies for main memory database clusters that are interposed between Internet applications and back-end databases as content caches. The task of management is to allocate data across individual cache databases and to route queries to the appropriate databases for execution. The goal is to maximize effective cache capacity and to minimize synchronization cost. We propose an affinity-based management system for main memory database cLUsters (*ALBUM*). ALBUM executes ea ...

Keywords: Main memory database, clustering, database administration, database cluster, file organization, query affinity, scalability

7 Network attached storage architecture

Garth A. Gibson, Rodney Van Meter

November 2000 **Communications of the ACM**, Volume 43 Issue 11

Publisher: ACM Press

Full text available: pdf(224.67 KB) Additional Information: full citation, references, citings, index terms

html(43.39 KB)

8 HLA-based Adaptive Distributed Simulation of Wireless Mobile Systems
Luciano Bononi, Gabriele D'Angelo, Lorenzo Donatiello
June 2003 **Proceedings of the seventeenth workshop on Parallel and distributed simulation**
Publisher: IEEE Computer Society
Full text available: pdf(367.43 KB) Publisher Site Additional Information: full citation, abstract, citings, index terms

Wireless networks' models differ from wired ones atleast in the innovative dynamic effects of host-mobilityand open-broadcast nature of the wireless medium.Topology changes due to simulated hosts' mobility mapon causality effects in the "areas of influence" of eachmobile device. The analysis of wireless networks ofinterest today may include a potentially high number ofsimulated hosts, resulting in performance and scalabilityproblems for discrete-event sequential simulation toolsand methods, on a ...

9 Hierarchial architecture for real-time adaptive resource management
Inout Cardei, Rakesh Jha, Mihaela Cardei, Allalaghatta Pavan
April 2000 **IFIP/ACM International Conference on Distributed systems platforms**
Publisher: Springer-Verlag New York, Inc.
Full text available: pdf(261.48 KB) Additional Information: full citation, abstract, references

This paper presents the Real Time Adaptive Resource Management system (RTARM [1]), developed at the Honeywell Technology Center. RTARM supports provision of integrated services for real-time distributed applications and offers management services for end-to-end QoS negotiation, QoS adaptation, real-time monitoring and hierarchical QoS feedback adaptation. In this paper, we focus on the hierarchical architecture of RTARM, its flexibility, internal mechanisms and protocols that enable ...

10 Dynamically Adaptive Partition-Based Data Distribution Management
Bora I. Kumova
June 2005 **Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation PADS '05**
Publisher: IEEE Computer Society
Full text available: pdf(146.21 KB) Additional Information: full citation, abstract

Performance and scalability of distributed simulations depends primarily on the effectiveness of the employed data distribution management (DDM) algorithm, which aims at reducing the overall computational and messaging effort on the shared data to a necessary minimum. Existing DDM approaches, which are variations and combinations of two basic techniques, namely region-based and grid-based techniques, perform purely in the presence of load differences. We introduce the partition-based technique t ...

11 High-performance sorting on networks of workstations
Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, David E. Culler, Joseph M. Hellerstein, David A. Patterson
June 1997 **ACM SIGMOD Record , Proceedings of the 1997 ACM SIGMOD international conference on Management of data SIGMOD '97**, Volume 26 Issue 2
Publisher: ACM Press
Full text available: pdf(1.53 MB) Additional Information: full citation, abstract, references, citings, index terms

We report the performance of NOW-Sort, a collection of sorting implementations on a Network of Workstations (NOW). We find that parallel sorting on a NOW is competitive to sorting on the large-scale SMPs that have traditionally held the performance records. On a 64-node cluster, we sort 6.0 GB in just under one minute, while a 32-node cluster finishes the Datamation benchmark in 2.41 seconds. Our implementations can be applied to a variety of disk, memory, and processor configura ...

12 Data management in networks: experimental evaluation of a provably good strategy

Christof Krick, Friedhelm Meyer auf der Heide, Harald Räcke, Berthold Vöcking, Matthias Westermann

June 1999 **Proceedings of the eleventh annual ACM symposium on Parallel algorithms and architectures**

Publisher: ACM Press

Full text available: pdf(1.52 MB) Additional Information: full citation, references, citings, index terms

13 Query evaluation techniques for large databases

Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

Publisher: ACM Press

Full text available: pdf(9.37 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

14 Architecture and algorithms for scalable mobile QoS

Bahareh Sadeghi, Edward W. Knightly

January 2003 **Wireless Networks**, Volume 9 Issue 1

Publisher: Kluwer Academic Publishers

Full text available: pdf(337.13 KB) Additional Information: full citation, abstract, references, index terms

Supporting Quality of Service (QoS) is an important objective for future mobile systems, and requires resource reservation and admission control to achieve. In this paper, we introduce an admission control scheme termed Virtual Bottleneck Cell, an approach designed to scale to many users and handoffs, while simultaneously controlling "hot spots". The key technique is to hierarchically control an aggregated virtual system, ensuring QoS objectives are satisfied in the underlying system without per ...

Keywords: QoS, adaptive clustering, admission control, mobile networks

15 Implementing global memory management in a workstation cluster

M. J. Feeley, W. E. Morgan, E. P. Pighin, A. R. Karlin, H. M. Levy, C. A. Thekkath

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5

Publisher: ACM Press

Full text available: pdf(1.52 MB) Additional Information: full citation, references, citings, index terms

16 The Panasas ActiveScale Storage Cluster - Delivering Scalable High Bandwidth Storage

Hong Tang, Aziz Gulbeden, Jingyu Zhou, William Strathearn, Tao Yang, Lingkun Chu

November 2004 **Proceedings of the 2004 ACM/IEEE conference on Supercomputing**
Publisher: IEEE Computer Society
Full text available: pdf(199.24 KB) Additional Information: full citation, abstract

Fundamental advances in high-level storage architectures and low-level storage-device interfaces greatly improve the performance and scalability of storage systems. Specifically, the decoupling of storage control (i.e., file system policy) from datapath operations (i.e., read, write) allows client applications to leverage the readily available bandwidth of storage devices while continuing to rely on the rich semantics of todayýs file systems. Further, the evolution of storage interfaces from blo ...

17 False sharing problems in cluster-based disk arrays
Hai Jin, Kai Hwang
February 1999 **Proceedings of the 1999 ACM symposium on Applied computing**
Publisher: ACM Press
Full text available: pdf(618.92 KB) Additional Information: full citation, references, index terms

Keywords: RAID, clusters of workstations, false sharing, storage system architecture

18 Libraries and applications: Locality aware dynamic load management for massively multiplayer games
Jin Chen, Baohua Wu, Margaret Delap, Björn Knutsson, Honghui Lu, Cristiana Amza
June 2005 **Proceedings of the tenth ACM SIGPLAN symposium on Principles and practice of parallel programming**
Publisher: ACM Press
Full text available: pdf(245.56 KB) Additional Information: full citation, abstract, references, index terms

Most massively multiplayer game servers employ static partitioning of their game world into distinct mini-worlds that are hosted on separate servers. This limits cross-server interactions between players, and exposes the division of the world to players. We have designed and implemented an architecture in which the partitioning of game regions across servers is transparent to players and interactions are not limited to objects in a single region or server. This allows a finer grain partitioning, ...

Keywords: adaptive, distributed, load balancing, locality aware, massively multiplayer games

19 Comparison of signaling loads for PCS systems
Thomas F. La Porta, Malathi Veeraraghavan, Richard W. Buskens
December 1996 **IEEE/ACM Transactions on Networking (TON)**, Volume 4 Issue 6
Publisher: IEEE Press
Full text available: pdf(1.72 MB) Additional Information: full citation, references, citings, index terms

20 Design and evaluation of parallel pipelined join algorithms
James P. Richardson, Hongjun Lu, Krishna Mikkilineni
December 1987 **ACM SIGMOD Record , Proceedings of the 1987 ACM SIGMOD international conference on Management of data SIGMOD '87**, Volume 16 Issue 3
Publisher: ACM Press
Full text available: pdf(1.21 MB) Additional Information: full citation, abstract, references, citings, index terms

The join operation is the most costly operation in relational database management systems. Distributed and parallel processing can effectively speed up the join operation. In this paper, we describe a number of highly parallel and pipelined multiprocessor join

algorithms using sort-merge and hashing techniques. Among them, two algorithms are parallel and pipelined versions of traditional sort-merge join methods, two algorithms use both hashing and sort-merge techniques, and another two are ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next